Direct Memory Access and DMA-controlled I/O

Introduction:

In this chapter, we provide examples and a detailed explanation of the DMA I/O technique used in personal computer systems including those using Intel family of microprocessors.

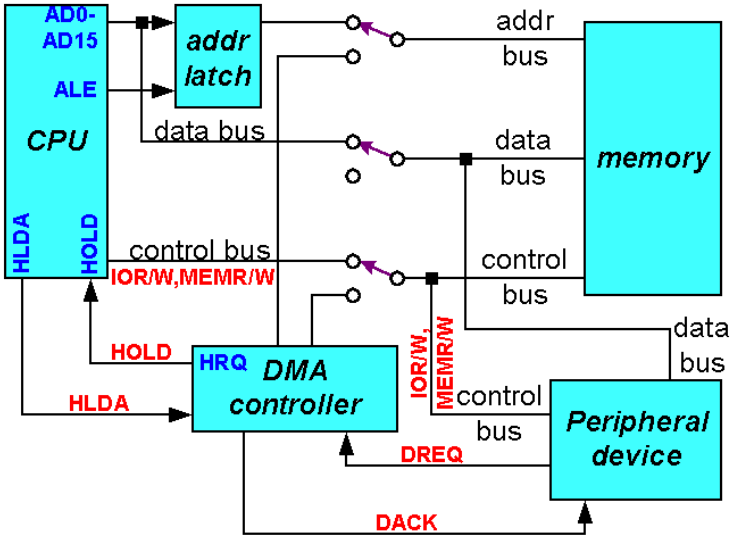
Objectives:

Upon completion of this chapter, you will be able to

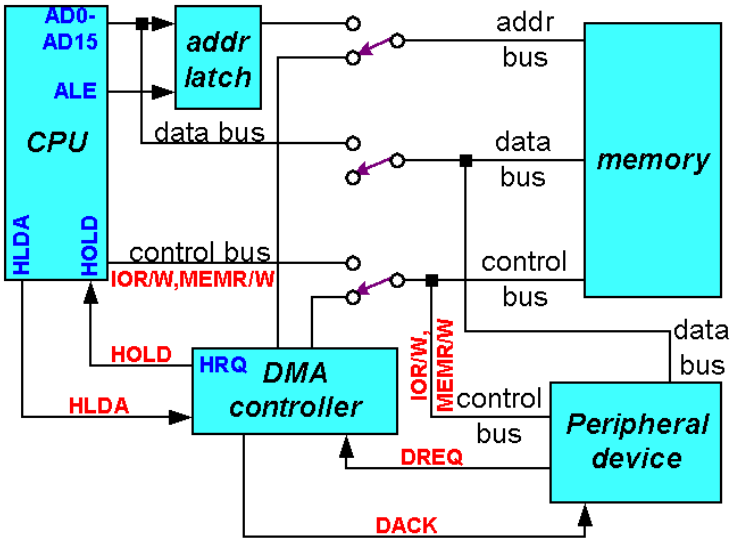
1. Describe a DMA transfer.
2. Explain the operation of the HOLD and HLDA direct memory access control signals.
3. Explain the function of the 8237 DMA controller when used for DMA transfers.
4. Explain the function of the 8289 bus arbiter.
5. Program the 8237 to accomplish DMA transfer.
6. Describe the disk standards found in personal computer.
7. Describe the various video interface standards that are found in the personal computer

1. Basic DMA operation

- The direct memory access (DMA) I/O technique provides direct access to the memory while the microprocessor is temporarily disabled.
- A DMA controller temporarily borrows the address bus, data bus, and control bus from the microprocessor and transfers the data bytes directly between an I/O port and a series of memory locations.
- The DMA transfer is also used to do high-speed memory-to-memory transfers.
- Two control signals are used to request and acknowledge a DMA transfer in the microprocessor-based system.
- The HOLD signal is a bus request signal which asks the microprocessor to release control of the buses after the current bus cycle.
- The HLDA signal is a bus grant signal which indicates that the microprocessor has indeed released control of its buses by placing the buses at their high-impedance states.
- The HOLD input has a higher priority than the INTR or NMI interrupt inputs.

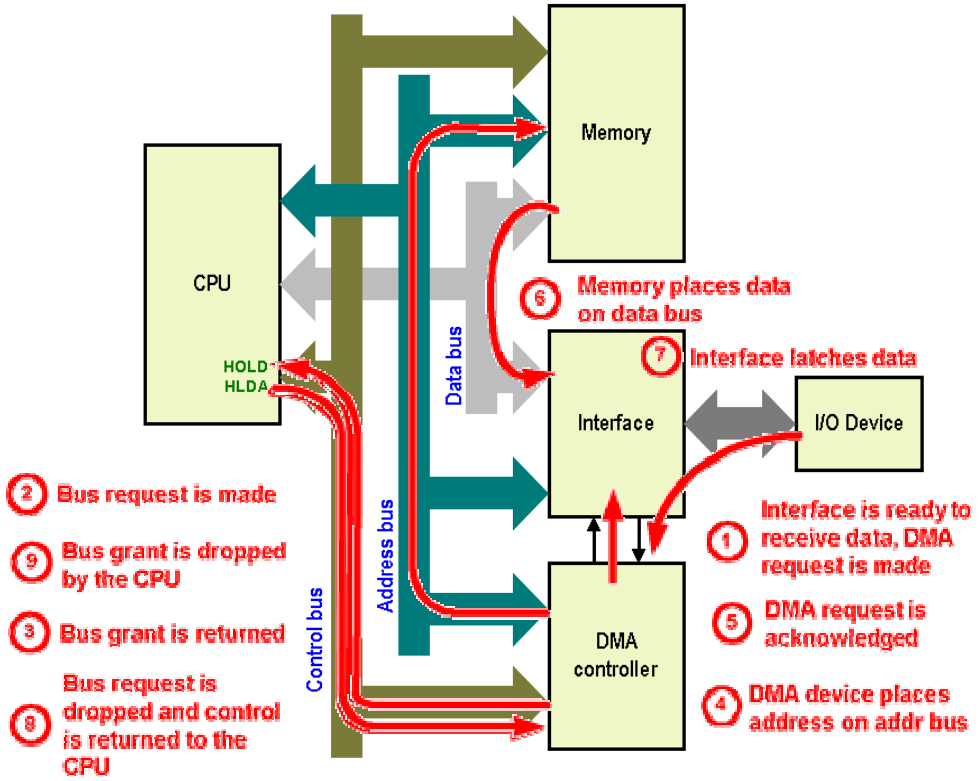


When DMA does not operate



When DMA operates

Example: memory-to-device transfer



2. The 8237 DMA controller

- The 8237 DMA controller supplies the memory and I/O with control signals and memory address information during the DMA transfer.
- The 8237 is a four-channel device that is compatible to the 8086/8088 microprocessors and can be expanded to include any number of DMA channel inputs.

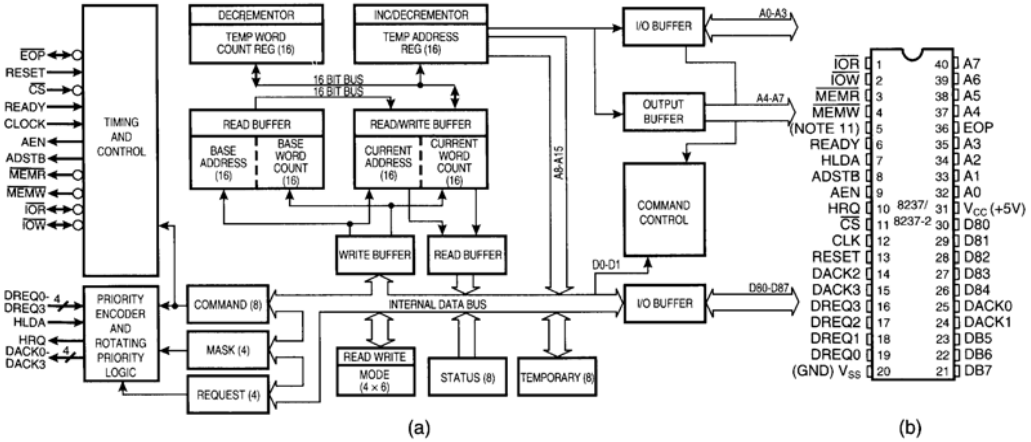
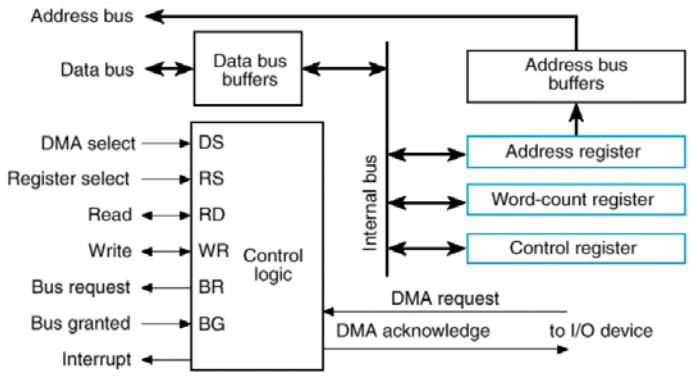


FIGURE 12–3 The 8237A–5 programmable DMA controller. (a) Block diagram, and (b) pin-out (Courtesy of Intel Corporation)



A conceptual DMA controller

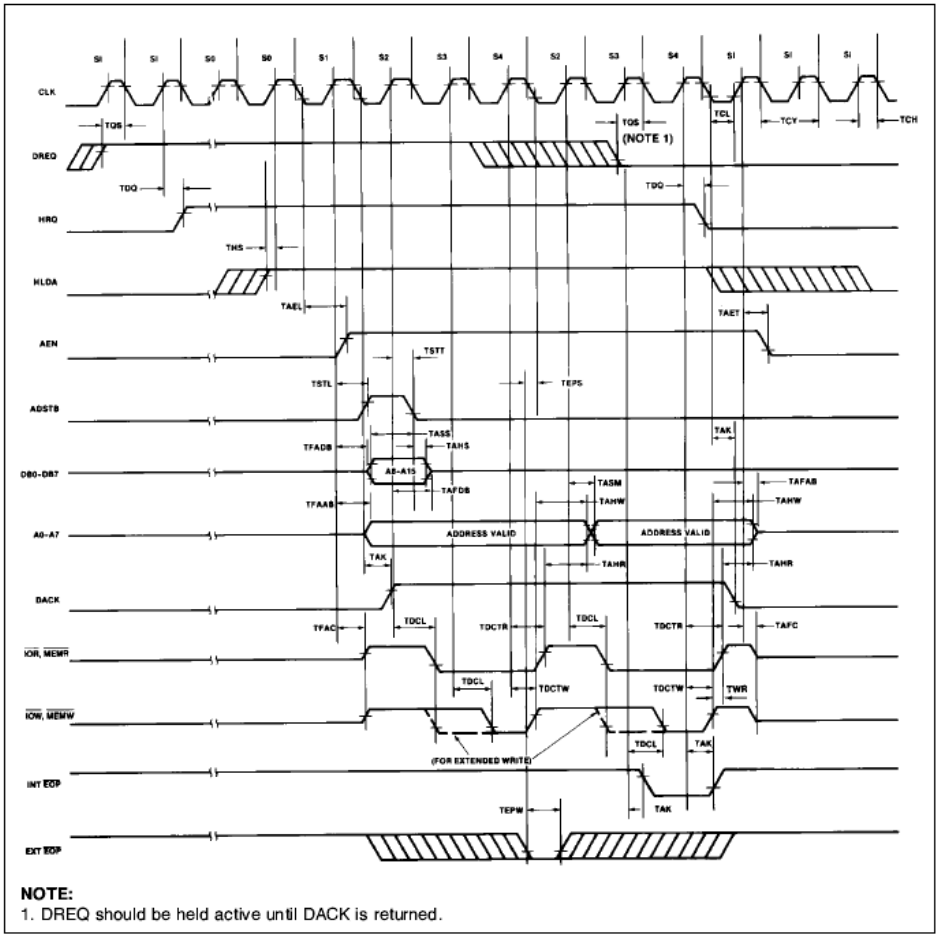
- The 8237 is capable of DMA transfers at rates of up to 1.6M bytes per second.
- Each channel is capable of addressing a full 64K-byte section of memory and can transfer up to 64K bytes with a single programming.

Some important signal pins:

- $DREQ_i$ (DMA request): Used to request a DMA transfer for a particular DMA channel.
- $DACK_i$ (DMA channel acknowledge): Acknowledges a channel DMA request from a device.
- HRQ (Hold request): Requests a DMA transfer.
- HLDA (Hold acknowledge) signals the 8237 that the microprocessor has relinquished control of the address, data and control buses.
- AEN (Address enable): Enables the DMA address latch connected to the 8237 and disable any buffers in the system connected to the microprocessor. (Use to take the control of the address bus from the microprocessor)
- ADSTB (Address strobe): Functions as ALE to latch address during the DMA transfer.
- $\overline{EOP}$ (End of process): Signals the end of the DMA process.
- $\overline{IOR}$ (I/O read): Used as an input strobe to read data from the 8237 during programming and used as an output strobe to read data from the port during a DMA write cycle.

- $\overline{IOW}$ (I/O write): Used as an input strobe to write data to the 8237 during programming and used as an output strobe to write data to the port during a DMA read cycle.
- $\overline{MEMW}$ (Memory write): Used as an output to cause memory to write data during a DMA write cycle.
- $\overline{MEMR}$ (Memory read): Used as an output to cause memory to read data during a DMA read cycle.

DMA TRANSFER TIMING



NOTE:
1. DREQ should be held active until DACK is returned.

Figure 11. DMA Transfer

Flow of control signals

8086		**8237**		**device**
			←	DREQ
	←	HREQ		
HLDA	→			
		Use AEN to take the control of the buses		
		DACK	→	
		Provide address plus $\overline{MEMW}$ $\overline{IOW}$ *or* $\overline{MEMR}$ $\overline{IOW}$		
		•		
		•		
	←	$\overline{EOP}$		
		HREQ ↓		
		AEN ↓		

Internal registers

- The ***current address register*** (CAR) is used to hold the 16-bit memory address used for the DMA transfer.
- The ***current word count register*** (CWCR) programs a channel for the number of bytes (up to 64K) transferred during a DMA action.
- The ***base address*** (BA) and ***base word count*** (BWC) registers are used when auto-initialization is selected for a channel. In this mode, their contents will be reloaded to the CAR and CWCR after the DMA action is completed.
- Each channel has its own CAR, CWCR, BA and BWC.
- The ***command register*** (CR) programs the operation of the 8237 DMA controller
- The ***mode register*** (MR) programs the mode of operation for a channel.
- The ***request register*** (RR) is used to request a DMA transfer via software, which is very useful in memory-to-memory transfers.
- The ***mask register set/reset*** (MRSR) sets or clears the channel mask to disable or enable particular DMA channels.
- The ***mask register*** (MSR) clears or sets all of the masks with one command instead of individual channels as with the MRSR.
- The ***status register*** (SR) shows the status of each DMA channel.

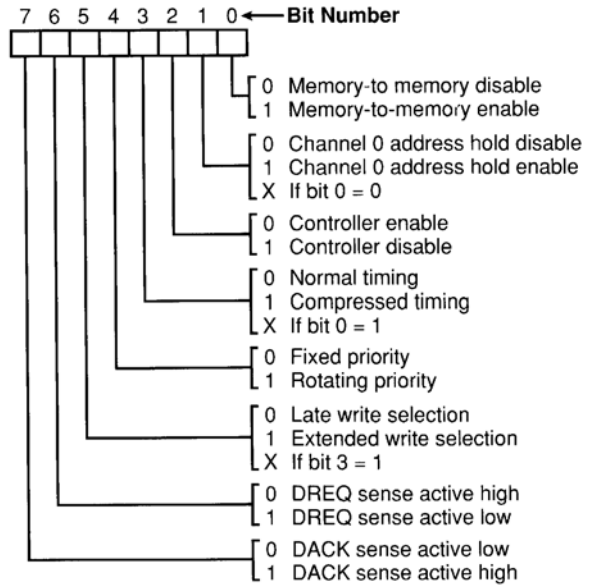


FIGURE 12–4 8237A–5 command register (Courtesy of Intel Corporation)

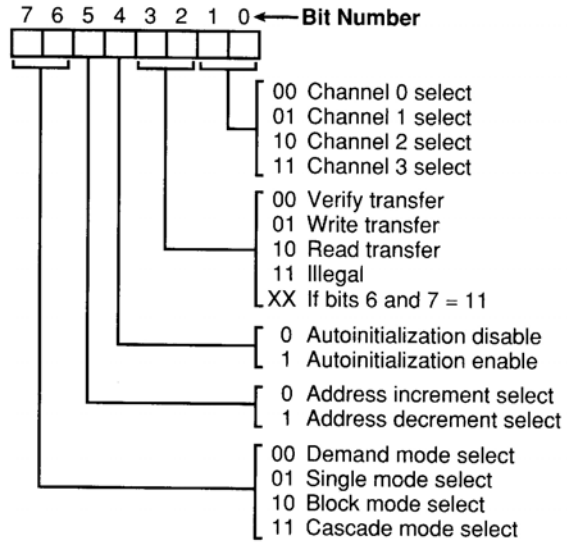


FIGURE 12–5 8237A–5 mode register (Courtesy of Intel Corporation)

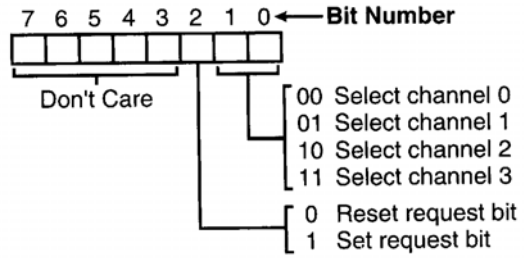


FIGURE 12–6 8237A–5 request register (Courtesy of Intel Corporation)

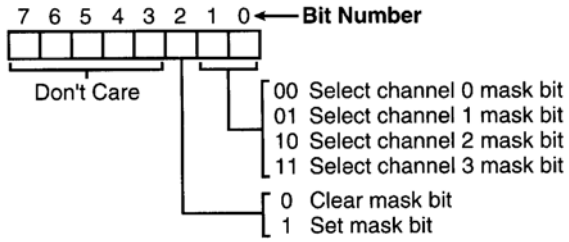


FIGURE 12–7 8237A–5 mask register set/reset mode (Courtesy of Intel Corporation)

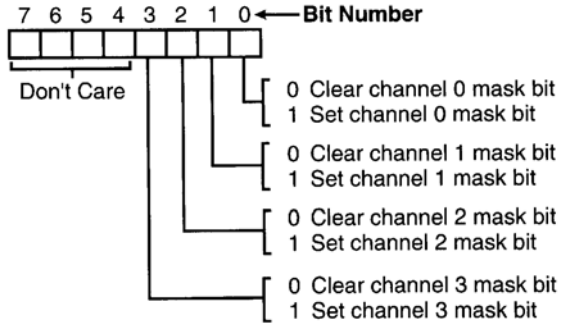


FIGURE 12–8 8237A–5 mask register (Courtesy of Intel Corporation)

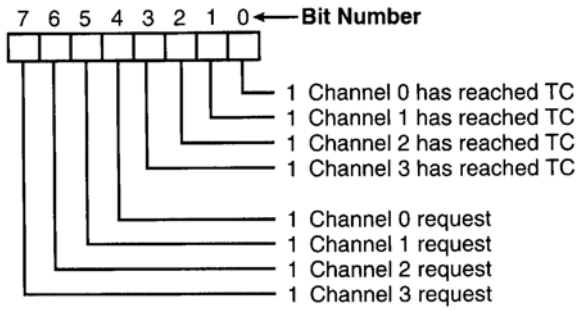


FIGURE 12–9 8237A–5 status register (Courtesy of Intel Corporation)

Channel	Register	Operation	Signals							Internal Flip-Flop	Data Bus DB0-DB7
			$\overline{CS}$	$\overline{IOR}$	$\overline{IOW}$	A3	A2	A1	A0		
0	Base and Current Address	Write	0	1	0	0	0	0	0	0	A0-A7
			0	1	0	0	0	0	0	1	A8-A15
	Current Address	Read	0	0	1	0	0	0	0	0	A0-A7
			0	0	1	0	0	0	0	1	A8-A15
	Base and Current Word Count	Write	0	1	0	0	0	0	1	0	W0-W7
			0	1	0	0	0	0	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	0	0	1	0	W0-W7
			0	0	1	0	0	0	1	1	W8-W15
1	Base and Current Address	Write	0	1	0	0	0	1	0	0	A0-A7
			0	1	0	0	0	1	0	1	A8-A15
	Current Address	Read	0	0	1	0	0	1	0	0	A0-A7
			0	0	1	0	0	1	0	1	A8-A15
	Base and Current Word Count	Write	0	1	0	0	0	1	1	0	W0-W7
			0	1	0	0	0	1	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	0	1	1	0	W0-W7
			0	0	1	0	0	1	1	1	W8-W15
2	Base and Current Address	Write	0	1	0	0	1	0	0	0	A0-A7
			0	1	0	0	1	0	0	1	A8-A15
	Current Address	Read	0	0	1	0	1	0	0	0	A0-A7
			0	0	1	0	1	0	0	1	A8-A15
	Base and Current Word Count	Write	0	1	0	0	1	0	1	0	W0-W7
			0	1	0	0	1	0	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	1	0	1	0	W0-W7
			0	0	1	0	1	0	1	1	W8-W15
3	Base and Current Address	Write	0	1	0	0	1	1	0	0	A0-A7
			0	1	0	0	1	1	0	1	A8-A15
	Current Address	Read	0	0	1	0	1	1	0	0	A0-A7
			0	0	1	0	1	1	0	1	A8-A15
	Base and Current Word Count	Write	0	1	0	0	1	1	1	0	W0-W7
			0	1	0	0	1	1	1	1	W8-W15
	Current Word Count	Read	0	0	1	0	1	1	1	0	W0-W7
			0	0	1	0	1	1	1	1	W8-W15

FIGURE 12–11 8237A–5 DMA channel I/O port addresses

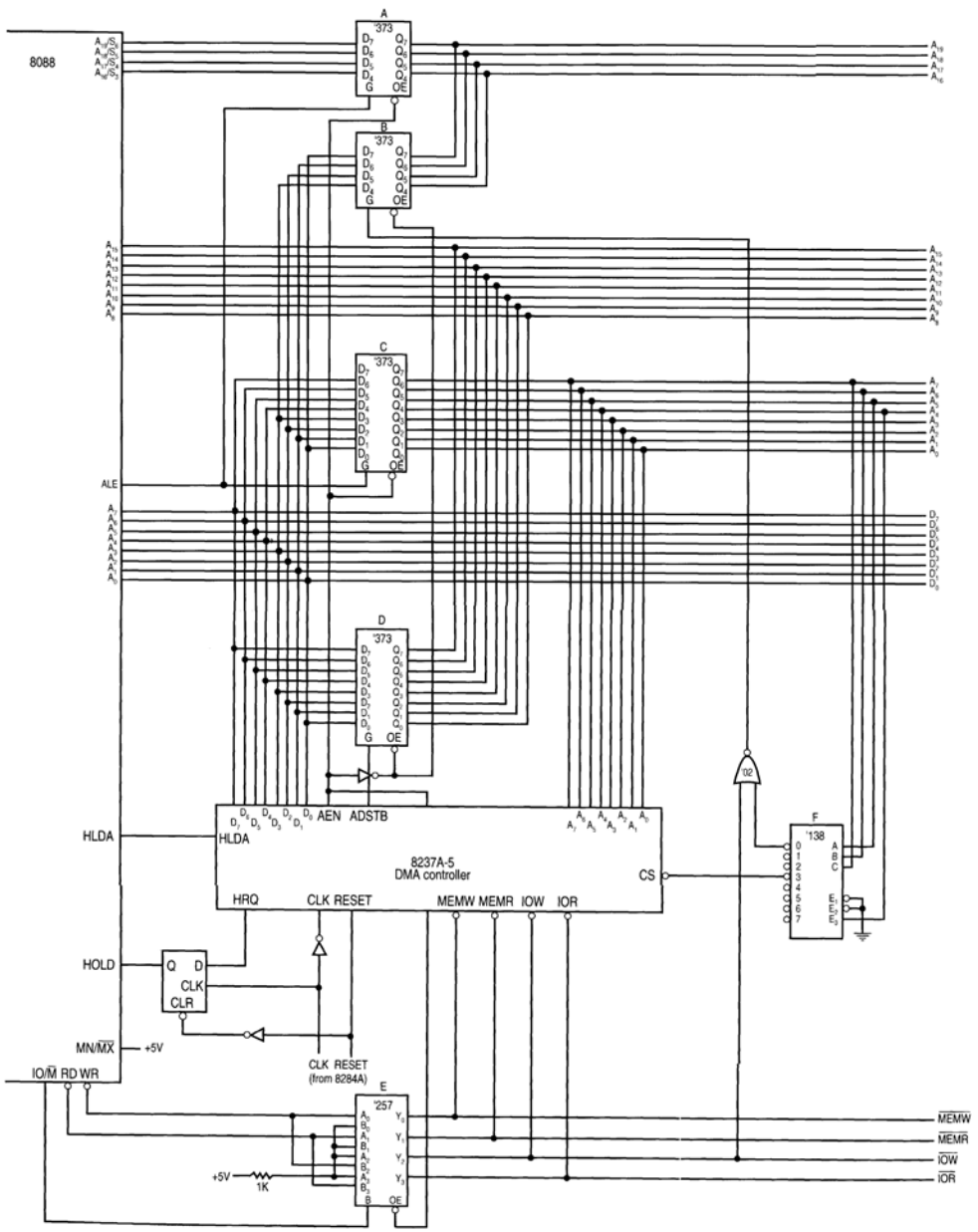
Software commands

- There are 3 software commands used to control the operation of the 8237.
  1. Clear the first/last (F/L) flip-flop
  2. Master clear - Disable all DMA channels
  3. Clear mask register - Enables all DMA channels

- The command is issued by writing a dummy byte to the corresponding port.

How an 8237 works in an 80X86-based system

FIGURE 12–12 Complete 8088 minimum mode DMA system (pp. 476–477)



- During normal 80x86 operation, the DMA controller and integrated circuits B and D are disabled.
- During a DMA action, IC A, C and E are disabled so that the 8237 can take control of the system through the address, control and data buses.

Programming the 8237

- There are 4 steps required to program the address and count registers first:
  1. Clear the F/L flip-flop with a clear F/L command
  2. Disable the channel
  3. Program the LSB and then MSB of the address
  4. Program the LSB and then MSB of the count

- Additional programming is required to select the mode of operation before the channel is enabled and started.

Sample memory-to-memory DMA transfer

EXAMPLE 12–1

```
                ;A procedure that transfers a block of data using the
                ;8237A DMA controller in Figure 12-12. This is a
                ;memory-to-memory block transfer.
                ;
                ;Calling parameters:
                ;   SI = source address
                ;   DI = destination address
                ;   CX = count
                ;   ES = segment of source and destination
;
= 0010          LATCHB  EQU   10H        ;latch B
= 007C          CLEAR_F EQU   7CH        ;F/L flip flop
= 0070          CH0_A   EQU   70H        ;channel 0 address
= 0072          CH1_A   EQU   72H        ;channel 1 address
= 0073          CH1_C   EQU   73H        ;channel 1 count
= 007B          MODE    EQU   7BH        ;mode
= 0078          CMMD    EQU   78H        ;command
= 007F          MASKS   EQU   7FH        ;masks
= 0079          REQ     EQU   79H        ;request register
= 0078          STATUS  EQU   78H        ;status register

0000            TRANS   PROC  FAR USES AX

0001  8C C0             MOV   AX,ES      ;program latch B
0003  8A C4             MOV   AL,AH
0005  C0 E8 04          SHR   AL,4
0008  E6 10             OUT   LATCHB,AL
000A  E6 7C             OUT   CLEAR_F,AL ;clear F/L flip-flop

000C  8C C0             MOV   AX,ES      ;program source address
000E  C1 E0 04          SHL   AX,4
0011  03 C6             ADD   AX,SI      ;form source offset
0013  E6 70             OUT   CH0_A,AL
0015  8A C4             MOV   AL,AH
0017  E6 70             OUT   CH0_A,AL

0019  8C C0             MOV   AX,ES      ;program destination address
001B  C1 E0 04          SHL   AX,4
001E  03 C7             ADD   AX,DI      ;form destination offset
0020  E6 72             OUT   CH1_A,AL
0022  8A C4             MOV   AL,AH
0024  E6 72             OUT   CH1_A,AL
0026  8B C1             MOV   AX,CX      ;program count
0028  48                DEC   AX         ;adjust count
0029  E6 73             OUT   CH1_C,AL
002B  8A C4             MOV   AL,AH
002D  E6 73             OUT   CH1_C,AL

002F  B0 88             MOV   AL,88H     ;program mode
0031  E6 7B             OUT   MODE,AL

0033  B0 85             MOV   AL,85H
0035  E6 7B             OUT   MODE,AL

0037  B0 01             MOV   AL,1       ;enable block transfer
0039  E6 78             OUT   CMMD,AL

003B  B0 0E             MOV   AL,0EH     ;unmask channel 0
003D  E6 7F             OUT   MASKS,AL

003F  B0 04             MOV   AL,4       ;start DMA transfer
0041  E6 79             OUT   REQ,AL

                        .REPEAT          ;wait until DMA complete
0043 E4 78              IN    AL,STATUS
                        .UNTIL AL &1

                        RET

                TRANS   ENDP
```


3. Shared-bus operation

- A *multiprocessing* system (also called distributed system) uses more than one microprocessor to accomplish the work.
- A *multitasking* system performs more than one task at a time.
- In a distributed, multiprocessing, multitasking environment, each microprocessor accesses two buses: (1) the local bus and (2) the remote or shared bus.
- The *local bus* is connected to memory and I/O devices that are directly accessed by a single microprocessor without any special protocol or access rules.
- The *shared bus* contains memory and I/O that are accessed by any microprocessor in the system.

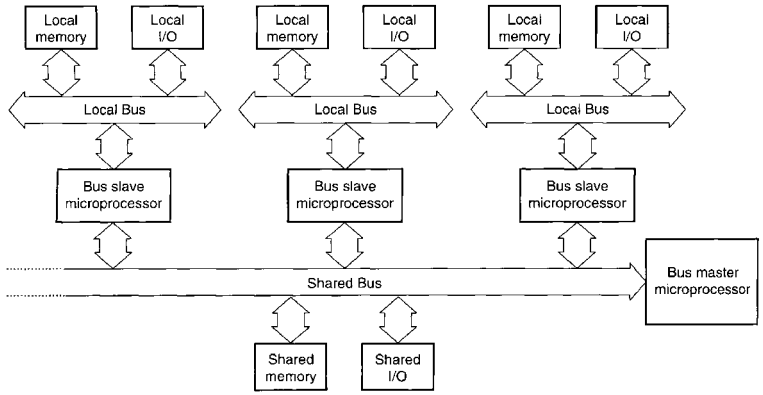


FIGURE 12–14 A block diagram illustrating the shared and local buses

- Characteristics of buses:

  Local bus

  - Resident to the microprocessor
  - Contains the resident or local memory and I/O

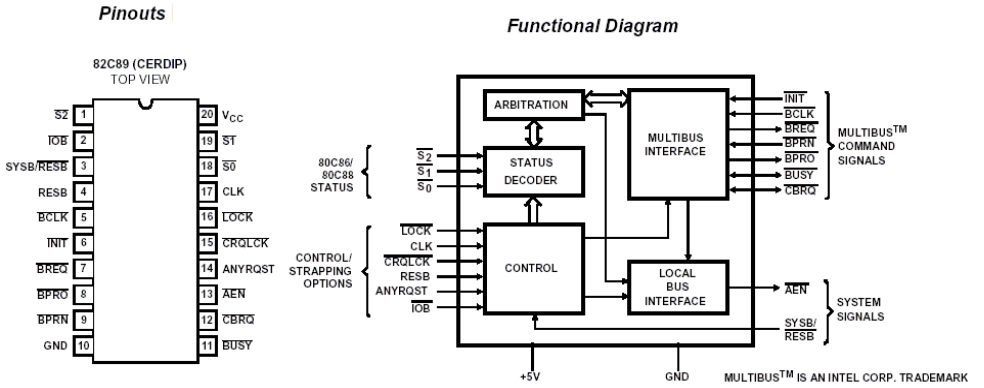
Shared bus

- Is connected to all microprocessors in the system
- Is used to exchange data between microprocessors in the system

- The shared bus in the personal computer is what we often call the local bus in the personal computer as it is local to the microprocessor in the personal computer.
- A *bus master* is a device (microprocessor or otherwise) that can control a bus containing memory and I/O.
- A remote bus master microprocessor can execute variable software but the DMA controller can only transfer data.
- Access to the shared bus for the remote bus master is accomplished via a bus arbiter.
- A *bus arbiter* functions to resolve priority between bus masters and allows only one device at a time to access the shared bus.

Bus arbiter

- The 8289 bus arbiter controls the interface of a bus master to a shared bus.
- The 8289 is designed to function with the 8086/8088 microprocessors.



Definition of some pins

- $\overline{BCLK}$ used to synchronize all shared-bus masters.
- $\overline{BPRN}$ allows the 8289 to acquire the shared bus on the next falling edge of the $\overline{BCLK}$ signal.
- $\overline{BPRO}$ used to resolve priority in a system that contains multiple bus masters.
- $\overline{BREQ}$ used to request access to the shared bus
- $\overline{BUSY}$ indicates that an 8289 has acquired the shared bus when used as an output , or used to detect that another 8289 has acquired the shared bus when used as an input.
- $\overline{CBRQ}$ a lower priority microprocessor is asking for the use of the shared bus.
- $\overline{IOB}$ selects, when $RESB$=1, whether the 8289 operates in a shared-bus system with I/O (=0) or with memory and I/O (==1)

- *RESB* configure the 8289 as a shared-bus master (=1) or a local-bus master (=0).
- *SYSB* / $\overline{RESB}$ selects the shared-bus (=1) or the resident local bus (=0)

General 8289 operation

MODE	PIN STRAPPING
Single Bus Multi-Master Mode	$\overline{IOB}$ = High RESB = Low
RESB Mode Only	$\overline{IOB}$ = High RESB = High
IOB Mode Only	$\overline{IOB}$ = Low RESB = Low
IOB Mode RESB Mode	$\overline{IOB}$ = Low RESB = High

Three basic operation modes of an 8289:

(1) I/O peripheral bus mode: All devices on the local bus are treated as I/O, including memory, and are accessed by I/O instructions. The shared bus is assessed by memory access.

(2) Resident bus mode: Allows memory and I/O accesses on both the local and shared bus.

(3) Single-bus mode: Cannot access local memory and local I/O

Example: 8088 operates in the remote mode with the local and shared bus connection

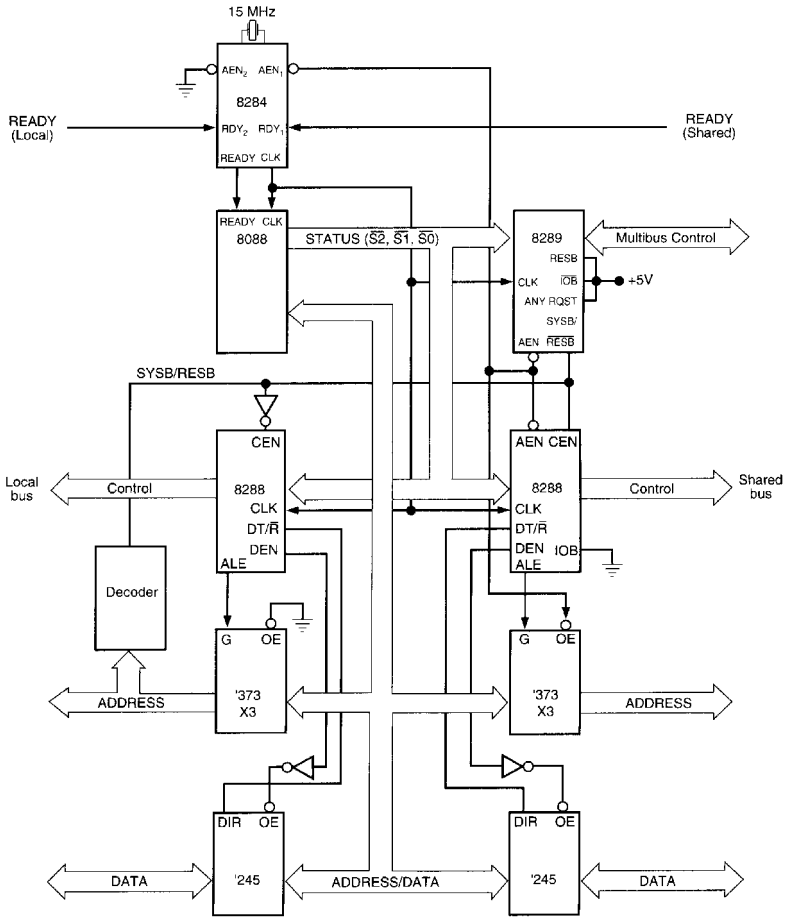


FIGURE 12–15 The 8088 operated in the remote mode illustrating the local and shared bus connections

- The 8288 is used as a bus controller when the 8088 operates in MAX mode.
- The shared bus is only to pass information from one processor to another.

- The bus masters function in their own local bus modes using their own local programs, memory, and I/O space.
- Microprocessors connected in a system like this is called *parallel* or *distributed processors* as they can execute software in parallel.
- The shared bus is mapped to some particular address locations such that accessing these address locations implies accessing the shared bus.
- The address decoder can detect the intention of accessing the shared bus and then activates the corresponding 8288 and configures the 8289 via 8289's *RESB* input pin.
- Blocking occurs whenever another microprocessor is accessing the shared bus.
- The 8289 controls the shared bus by making the READY input to the microprocessor be 0 if access to the shared bus is denied.
- Wait states are added until READY is 1.

Example: System illustrating single-bus and resident-bus connections

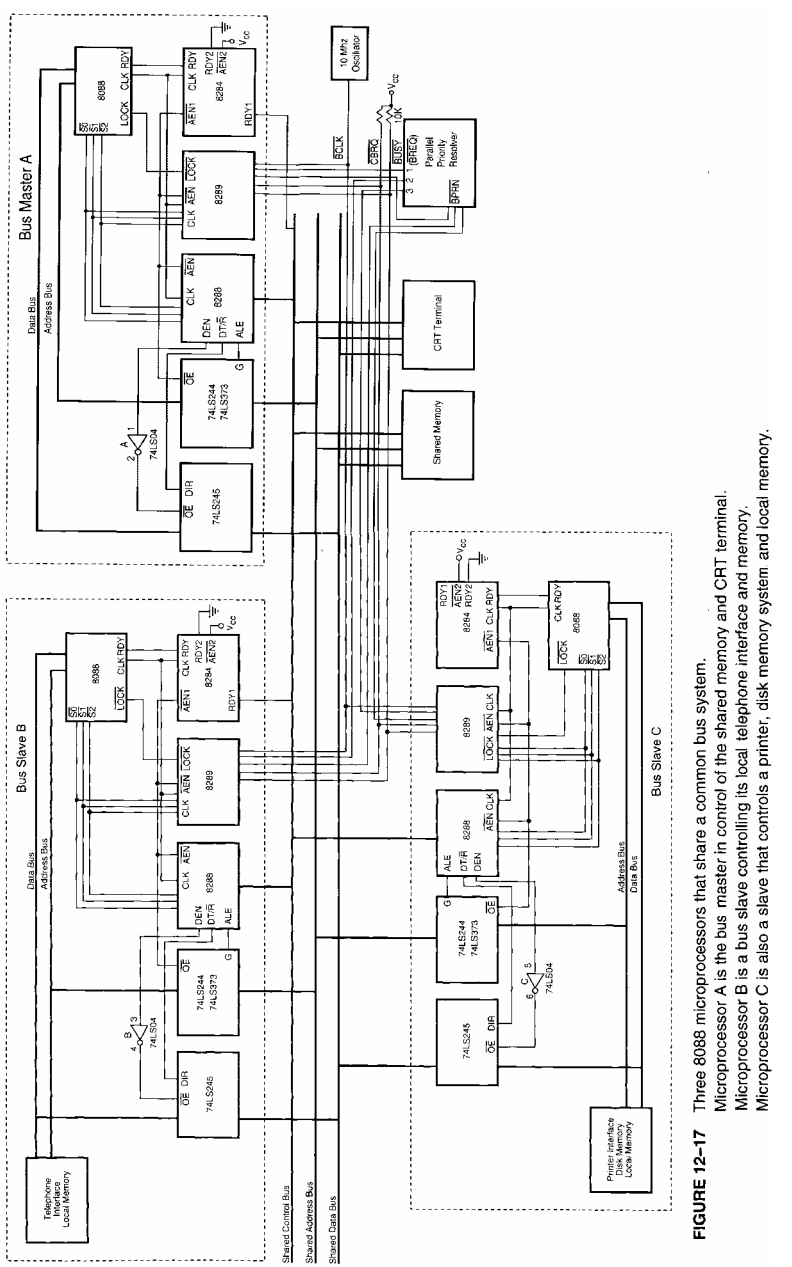


FIGURE 12–17 Three 8088 microprocessors that share a common bus system.
Microprocessor A is the bus master in control of the shared memory and CRT terminal.
Microprocessor B is a bus slave controlling its local telephone interface and memory.
Microprocessor C is also a slave that controls a printer, disk memory system and local memory.

- Microprocessor A is referred to as system bus master as it is responsible for coordinating the main memory and I/O task.

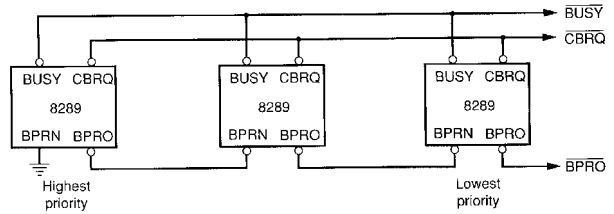
Micro-processor	Bus Arbiter's Mode	Function
A	Single bus mode (No local bus)	Allows the user to execute programs and control the system via a CRT terminal.
B	Resident-bus mode (Can access to both the shared bus and their own local buses)	Handles telephone communications and passes the information to the shared memory.
C		Used as a print spooler.

Priority logic using the 8289

- Only one can access the shared bus at a time.
- Two methods can be used to solve the priority: the daisy-chain and the parallel-priority schemes.

Daisy-chain priority

FIGURE 12–18 Daisy-chain 8289 priority resolver



- If no requests are active, all $\overline{BPRN}$ inputs will see 0.
- When an 8289 initiates a request, as soon as it receives a bus acknowledgement, its $\overline{BPRO}$ becomes 1 and blocks all lower priority 8289s.
- Potential problem arise when more than 1 microprocessor access the shared bus at a time.
- It's use is limited to no more than 3 8289s in a system that uses a bus clock of less than 10 MHz

Parallel priority

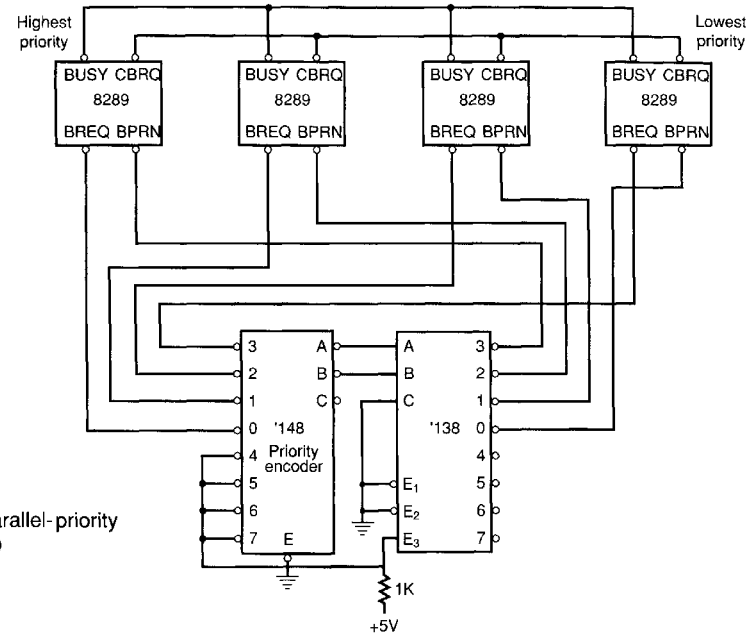


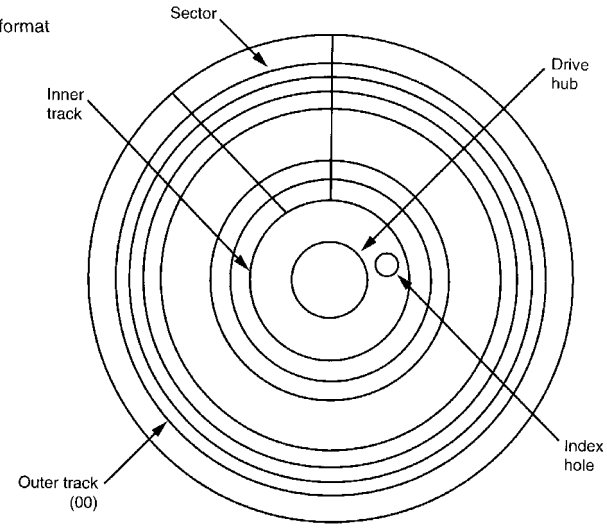
FIGURE 12–19 Parallel-priority resolver for the 8289

- If all 8289 arbiters are idle(SYSB/$\overline{RESB}$=0), the highest priority 8289 will gain access to the shared bus if it is requested by its microprocessor.
- If a lower priority request is made,
  - The $\overline{BREQ}$ output becomes 0, which causes the priority encoder to place a 0 on the corresponding 8289's $\overline{BREQ}$ input and allows access to the shared bus.
  - The $\overline{BUSY}$ signal becomes 0 and locks out any other request.
- If simultaneous requests occur, the 74LS148 automatically resolves the priority to prevent conflicts

4. Disk memory systems

Floppy disk memory

FIGURE 12–26 The format of a 5¼″ floppy disk

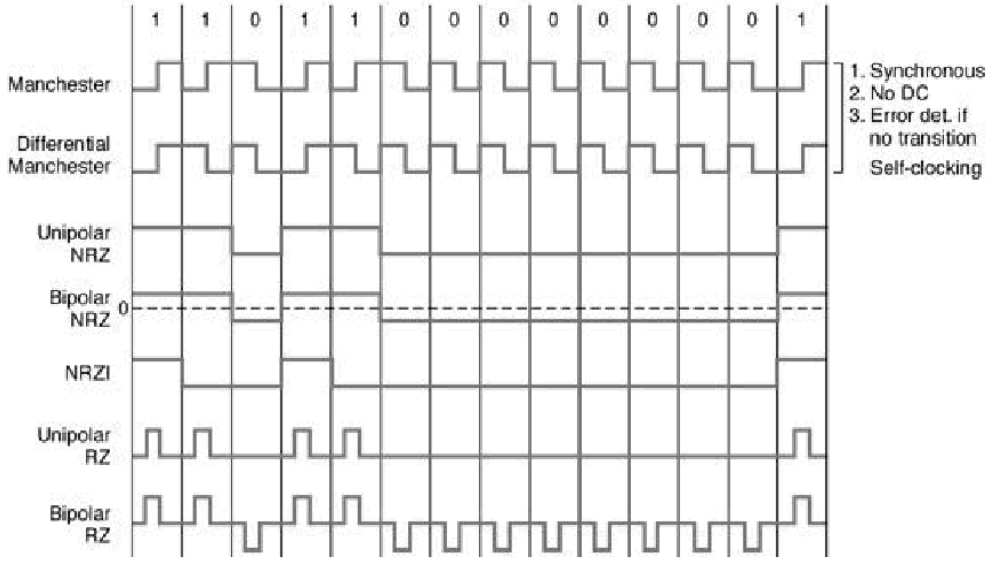


Medium	Double/ single sided	Sectors/ Track	Track/ side	Size
3.5"	D	18	80	1.44 Mbyte
3.5"	D	9	80	720 kbyte
5.25"	D	15	80	1.2 Mbyte
5.25"	S	8	80	320 kbyte
3.5"	S	8	80	320 kbyte
5.25"	D	8	80	640 kbyte
3.5"	D	8	80	640 kbyte
5.25"	S	9	40	180 kbyte
5.25"	D	9	40	360 kbyte
8"	D	26	77	1.96 Mbyte
5.25"	S	8	40	160 kbyte
8"	S	2	77	77 kbyte
8"	S	6	77	231 kbyte
8"	S	8	77	308 kbyte
5.25"	D	8	40	320 kbyte

Table 4-1. Floppy disk parameters

- A *track* is a concentric ring of data stored on a surface of a disk.
- A *sector* is a common subdivision of a track that is designed to hold a reasonable amount of data.
- A set of tracks is called a *cylinder* and consists of 1 top and 1 bottom track.

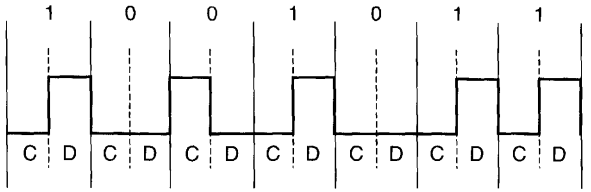
- The magnetic recording technique used is *non-return to zero* (NRZ) recording.



Manchester: transition at center of each cell. 1 = + trans. 0 = – trans.
Differential Manchester: transition at center of each cell. 1 = no trans. at beginning of cell. 0 = trans. at beginning of cell.
Unipolar NRZ: stays + and does not return to 0 during binary 1 cell.
Bipolar NRZ: + and – levels and does not return to 0 in cell.
NRZI: Change in signal level at beginning of bit = 1 and vice versa.
Unipolar RZ: goes + and returns to 0 from 1 during cell time.
Bipolar RZ: 2 non-zero voltage levels. 1 = +, 0 = – . (See chapter 12.)

- Advantages of NRZ recoding:
  - It automatically erases old information when new information is recorded.
  - It ensures that information will not be affected by noise as the amplitude of the magnetic field contains no information.
- Data are stored in the form of MFM (*modified frequency modulation*) in modern floppy disk systems.

FIGURE 12–29 Modified frequency modulation (MFM) used with disk memory



- Rules of MFM:
  1. A data pulse is always stored for a logic 1.
  2. No data and no clock are stored for the 1st logic 0 in a string of logic 0's.
  3. The 2nd and subsequent logic 0's in a row contain a clock pulse, but no data pulse.

     The insertion of clock information in rule 3 is to maintain synchronization as data are read from the disk.

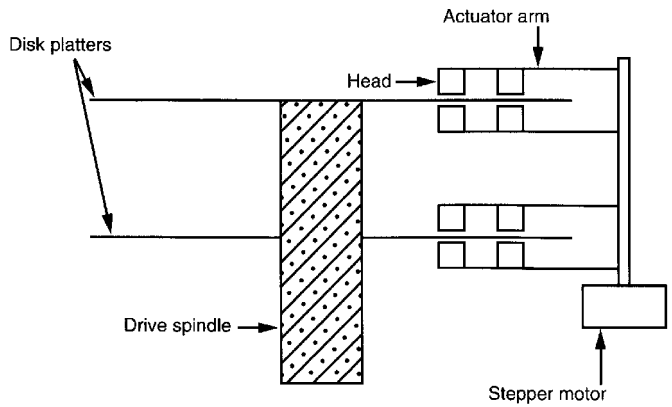
Hard disk memory

- It is also called *fixed disk* as it is not removable like the floppy disk.
- Difference between the floppy disk and the hard disk:

The hard disk memory uses a flying head (not touch the surface of the disk) to store and read data from the surface of the disk.

- Parking the flying head is necessary so as to avoid a head crash when it is not in use.
- The number of heads and disk surfaces are different in HD and FD.

FIGURE 12–31 A hard disk drive that uses four heads per platter



- Hard disk drives often store information in sectors that are 512 bytes long.
- Data are addressed in clusters of eight or more sectors on most hard disk drives.
- Hard disk drives use either MFM or RLL to store information
- Run-length limited (RLL)
  - RLL means that the run of zeros in a row is limited.

  - RLL 2,7 (the run of zeros is between 2 and 7) is commonly used.
  - Data are first encoded with table 12-2 before being stored.

TABLE 12–2
Standard RLL 2,7 coding

Input Data Stream	*RLL Output*
000	000100
10	0100
010	100100
0010	00100100
11	1000
011	001000
0011	00001000

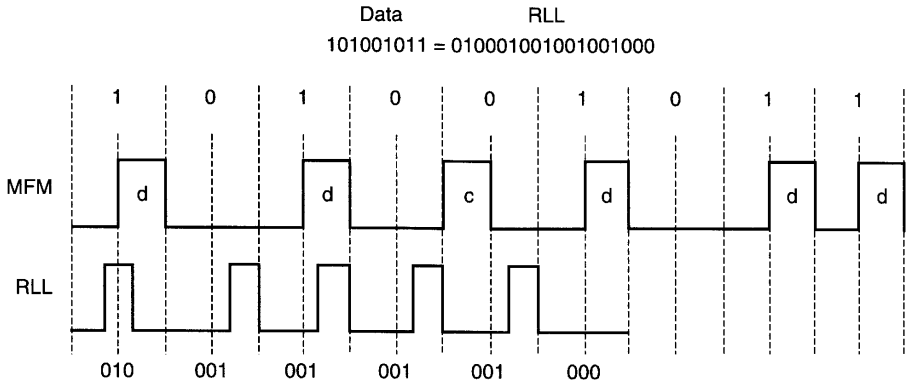


FIGURE 12–32 A comparison of MFM with RLL using data 101001011

- As compared with a MFM drive, a RLL drive can hold more information (up to 50% more) and be accessed at a higher rate.
- There are a number of *disk drive interfaces*: ST-506(obsolete), ESDI(obsolete), SCSI, IDE.
- *Small computer system interface* (SCSI): allows up to 7 different disk or other interfaces to be connected to the computer through same interface controller.

- *Integrated drive electronics* (IDE): IDE allows many disk drives to be connected to a system without worrying about the bus conflicts or controller conflicts.

5. Video displays

- Modern video displays can be either color or monochrome displays.
- Monochrome displays usually display information in amber, green or paper-white.
- Composite video displays work as a TV set and are disappearing because the resolution available is too low.

Video signals

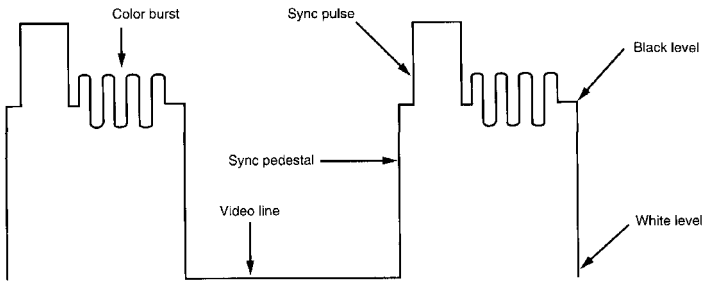


FIGURE 12–34 The composite video signal

- Composite video signals include not only video but sync pulses, sync pedestals, and a color burst.
- Composite video signals were designed to emulate television signals so that a home television receiver could function as a video monitor.
- Most modern video systems use direct video signals that are generated with separate sync signals.

- A monochrome monitor uses one wire for video, one for horizontal sync and one for vertical sync.
- A RGB monitors uses 3 video signals for deliver red, green and blue video information.

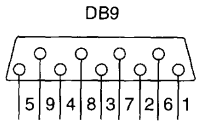
TTL RGB monitor

- The RGB monitor is available as either an analog or TTL monitor.
- It uses TTL level signals (0 or 5v) as video inputs and a 4th line called intensity to allow a change in intensity.
- The RGB video TTL display can display a total of 16 different colors.
- It is used in the CGA (color graphics adapter) system.

TABLE 12–3 16 colors found in the CGA display

Intensity	*Red*	*Green*	*Blue*	*Color*
0	0	0	0	Black
0	0	0	1	Blue
0	0	1	0	Green
0	0	1	1	Cyan
0	1	0	0	Red
0	1	0	1	Magenta
0	1	1	0	Brown
0	1	1	1	White
1	0	0	0	Gray
1	0	0	1	Bright Blue
1	0	1	0	Bright Green
1	0	1	1	Bright Cyan
1	1	0	0	Bright Red
1	1	0	1	Bright Magenta
1	1	1	0	Yellow
1	1	1	1	Bright White

FIGURE 12–35 The 9-pin connector found on a TTL monitor

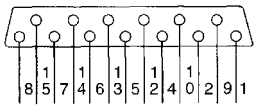


Pin	Function
1	Ground
2	Ground
3	Red video
4	Green video
5	Blue video
6	Intensity
7	Normal video
8	Horizontal retrace
9	Vertical retrace

- Pin normal video is used on a monochrome monitor for the luminance or brightness signal.

Analog RGB monitor

- Analog RGB monitors can display more than 16 colors as their video signal is analog (0-7V).
- Depending on the standard, the number of colors can be 256K, 16M or 24M.



Pin	Function
1	Red video
2	Green video (monochrome video)
3	Blue video
4	Ground
5	Ground
6	Red ground
7	Green ground (monochrome ground)
8	Blue ground
9	Blocked as a key
10	Ground
11	Color detect (ground on a color monitor)
12	Monochrome detect (ground on a monochrome monitor)
13	Horizontal retrace
14	Vertical retrace
15	Ground

FIGURE 12–36 The 15-pin connector found on an analog monitor

- No hole exists on the female connector for pin 9, which is used as a key.
- A digital-to-analog converter (DAC) is required to generate each color video voltage.
- Using a DAC of 6-bit(8-bit) resolution results in 256K (16M) colors.

Example: A circuit generating VGA(variable graphics array) video

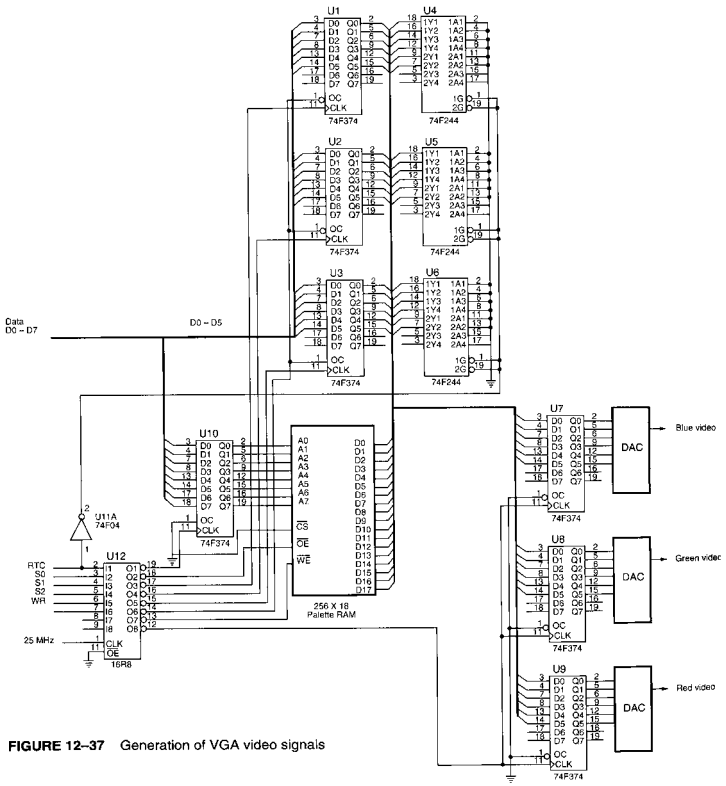
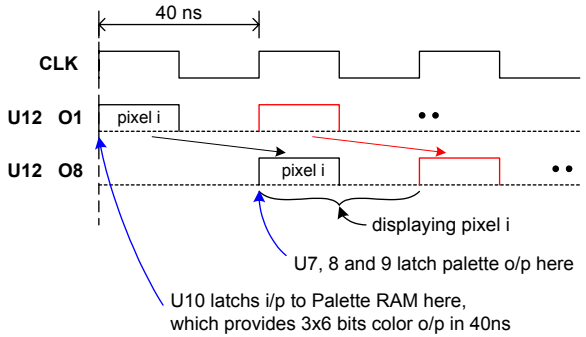


FIGURE 12–37 Generation of VGA video signals

- This system allows 256 colors out of a possible 256K colors to be displayed at one time.
- This system uses
  - 3x 6-bit DACs to generate analog video signals.
  - A high-speed palette SRAM (access time <40ns) to store 256 different 18-bit codes that represent 256 different colors.
  - 25 MHz clock for synchronization.

- <u>RTC=0 (display)</u>



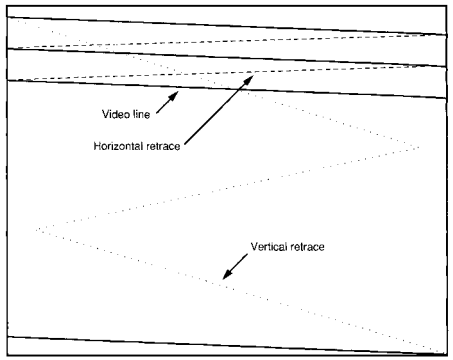
- <u>RTC=1 (Retrace)</u>
- During retrace, the video signal voltage sent to the display must be 0V.
- U4-U6 buffers are enabled so the inputs to the DACs are all 0.
- The palette can be modified during retrace as this will not disrupt the image displayed on the monitor.
- Modification of the palette:

- Load the 3 components of the color to U1, U2 and U3.
- Load the palette index of the color into U10.
- Generate a write pulse to $\overline{WE}$ of the palette RAM.

- The *resolution* of the display determines the amount of memory required for the video interface card.

- A *raster line* is the horizontal line of video information that is displayed on the monitor.
- A *pixel* is the smallest subdivision of a horizontal line.

FIGURE 12–38 A video screen illustrating the raster lines and retrace



- Timing issue:

  Consider a 640×480 display with

  ➢ Horizontal retrace repetition rate = 31500 Hz

  ➢ Vertical retrace repetition rate = 70.1 Hz

Time for scanning a line = 1/31500 = 31.75 μs

The pixel is 40 ns wide as a 25 MHz clock is used in this system.

Generating 640 pixels across one line takes 640×40ns = 25.6 μs ⇒ Time for H retrace = 1/31500 - 25.6 = 31.75-25.6 = 6.15 μs

Only 400 lines are displayed in a page ⇒ Time for V retrace = (31500/70.1 - 400) ×31.75 μs = 1.57 ms

- Parameters of some other display units:
  - SVGA : 800x600, ideal for 14" color monitor
  - EVGA(XVGA): 1024x768, ideal for a 21" or 25" monitor used in CAD systems
  - Home TV: ~400x300

- The use of palette limits the number of colors displayed at a time.
- The number of scanning lines is adjustable by changing the vertical and horizontal scanning rates.
- The vertical rate must be ≥ 50 Hz to avoid flicking.
- The horizontal and vertical scanning rates cannot be too high as the frequency of the signal applied to the coils in the yoke is limited.

- In a multi-sync monitors, the deflection coil is taped so that it can be driven with different deflection frequencies.
- Non-interlaced vs. interlaced system:
  - In the interlaced system, the video image is displayed by drawing half the image first with all the odd scanning lines, then the other half with the even scanning lines.
  - The quality of a non-interlaced system is higher.